**Content**

[1. TODO: To successfully pass the exam 2](#_Toc21176955)

[2. SageMaker built-in Algorithms -overview 3](#_Toc21176956)

[1. Common parameters 3](#_Toc21176957)

[2. What is the meaning of “downstream” processing? 3](#_Toc21176958)

[3. Algorithms one by one 4](#_Toc21176959)

[1. BlazingText 4](#_Toc21176960)

[2. DeepAR Forecasting 4](#_Toc21176961)

[3. Factorization Machines 4](#_Toc21176962)

[4. Image Classsification 4](#_Toc21176963)

[5. IP Insights 4](#_Toc21176964)

[6. k-means 4](#_Toc21176965)

[7. k-NN 4](#_Toc21176966)

[8. LDA 4](#_Toc21176967)

[9. Neural Topic Model 4](#_Toc21176968)

[10. Object2Vec 4](#_Toc21176969)

[11. Object Detection 4](#_Toc21176970)

[12. PCA 4](#_Toc21176971)

[13. RFC 4](#_Toc21176972)

[14. Semantic Segmentation 4](#_Toc21176973)

[15. Seq2Seq modelling 4](#_Toc21176974)

[16. XGBoost 4](#_Toc21176975)

1. TODO: To successfully pass the exam
2. LA videos
3. Community slack pinned staff.
4. **SageMaker docs (900 pages ?!) - algorithm chapter.**
5. Sagemaker FAQS
6. Example questions w links
7. Practice exam
8. Copy practice exam question to google and fins similar
9. Youtube ML vid list

* <https://www.youtube.com/playlist?list=PLhr1KZpdzukeKUChZG8SdYipk7INwphQm>
* <https://www.youtube.com/playlist?list=PLhr1KZpdzukdExxUr05-WIpdVYPWjgi43>

1. SageMaker built-in Algorithms -overview

Model and Inferences:

A machine learning algorithm uses example data to **create a generalized solution** (a **model**) that addresses the business question you are trying to answer. After you create a model using example data, you can use it to **answer the same business question for a new set of data**. This is also referred to as obtaining **inferences**.

* 1. Common parameters

Algorithms that are **parallelizable** can be deployed on multiple compute instances for distributed training. For the **Training Image and Inference Image Registry Path** column, use the :1 version tag to ensure that you are using a stable version of the algorithm. You can reliably host a model trained using an image with the :1 tag on an inference image that has the :1 tag. Using the :latest tag in the registry path provides you with the most up-to-date version of the algorithm, but might cause problems with backward compatibility. **Avoid using the :latest tag for production purposes.**

**Training Image and Inference Image Registry Path**

Use the paths and training input mode as follows:

* To create a **training job** (with a request to the CreateTrainingJob API), specify the Docker Registry path and the training input mode for the training image. You create a training job to train a model using a specific dataset.
* To **create a model** (with a CreateModel request), specify the Docker Registry path for the inference image. Amazon SageMaker launches machine learning compute instances that are based on the endpoint configuration and deploys the model, which includes the artifacts (the result of model training).

XGBoost implements an open-source algorithm that has been optimized for CPU computation.

* 1. What is the meaning of “downstream” processing?

"Downstream tasks is what the field calls those supervised-learning tasks that utilize a pre-trained model or component"

* 1. What is the difference between Classification and Regression?

*Fundamentally, classification is about predicting a label and regression is about predicting a quantity.*

* **Predictive modeling** is about the problem of learning a mapping function from inputs to outputs called function approximation.
* C**lassification** is the problem of predicting a discrete class label output for an example.
* **Regression** is the problem of predicting a continuous quantity output for an example.
  + 1. Function Approximation

Predictive modeling is the problem of developing a model using historical data to make a prediction on new data where we do not have the answer. Predictive modeling can be described as the mathematical problem of approximating a mapping function (f) from input variables (X) to output variables (y). This is called the problem of function approximation. The job of the modeling algorithm is to find the best mapping function we can give the time and resources available

* + 1. Classification Predictive Modelling

Classification predictive modeling is the task of approximating a mapping function (f) from input variables (X) to discrete output variables (y).

* A classification problem requires that examples be classified into one of two or more classes.
* A classification can have real-valued or discrete input variables.
* A problem with two classes is often called a two-class or binary classification problem.
* A problem with more than two classes is often called a multi-class classification problem.
* A problem where an example is assigned multiple classes is called a multi-label classification problem.
  + 1. Regression Predictive Modelling
    2. Classification v Regression
* Classification is the task of predicting a discrete class label.
* Regression is the task of predicting a continuous quantity.
* A classification algorithm may predict a continuous value, but the continuous value is in the form of a probability for a class label.
* A regression algorithm may predict a discrete value, but the discrete value in the form of an integer quantity.
* Classification predictions can be evaluated using accuracy, whereas regression predictions cannot.
* Regression predictions can be evaluated using root mean squared error, whereas classification predictions cannot.
  + 1. Convert between Classification and Regression

Class 0: $0 to $49

Class 1: $50 to $100

1. Algorithms one by one
   1. BlazingText

sentiment analysis, named entity recognition, machine translation, etc. Text classification is an important task for applications that perform web searches, information retrieval, ranking, and document classification.

The hyperparameters for the BlazingText algorithm depend on which mode you use: Word2Vec (unsupervised) and Text Classification (supervised).

* 1. DeepAR Forecasting

The Amazon SageMaker DeepAR forecasting algorithm is a supervised learning algorithm for forecasting scalar (one-dimensional) time series using recurrent neural networks (RNN). Classical forecasting methods, such as autoregressive integrated moving average (ARIMA) or exponential smoothing (ETS), fit a single model to each individual time series. They then use that model to extrapolate the time series into the future.

**RMSE – root mean square error**

* 1. Factorization Machines

A factorization machine is a general-purpose supervised learning algorithm that you can use for both classification and regression tasks. It is an extension of a linear model that is designed to capture interactions between features within high dimensional sparse datasets economically. For example, in a click prediction system, the factorization machine model can capture click rate patterns observed when ads from a certain ad-category are placed on pages from a certain page-category. Factorization machines are a good choice for tasks dealing with high dimensional sparse datasets, such as click prediction and item recommendation.

The Amazon SageMaker implementation of factorization machines considers only **pair-wise** (2nd order) **interactions** **between** **features**.

* 1. Image Classsification
  2. IP Insights
  3. k-means
  4. k-NN
  5. LDA
  6. Neural Topic Model
  7. Object2Vec
  8. Object Detection
  9. PCA
  10. RFC
  11. Semantic Segmentation
  12. Seq2Seq modelling
  13. XGBoost